

FIG. 1

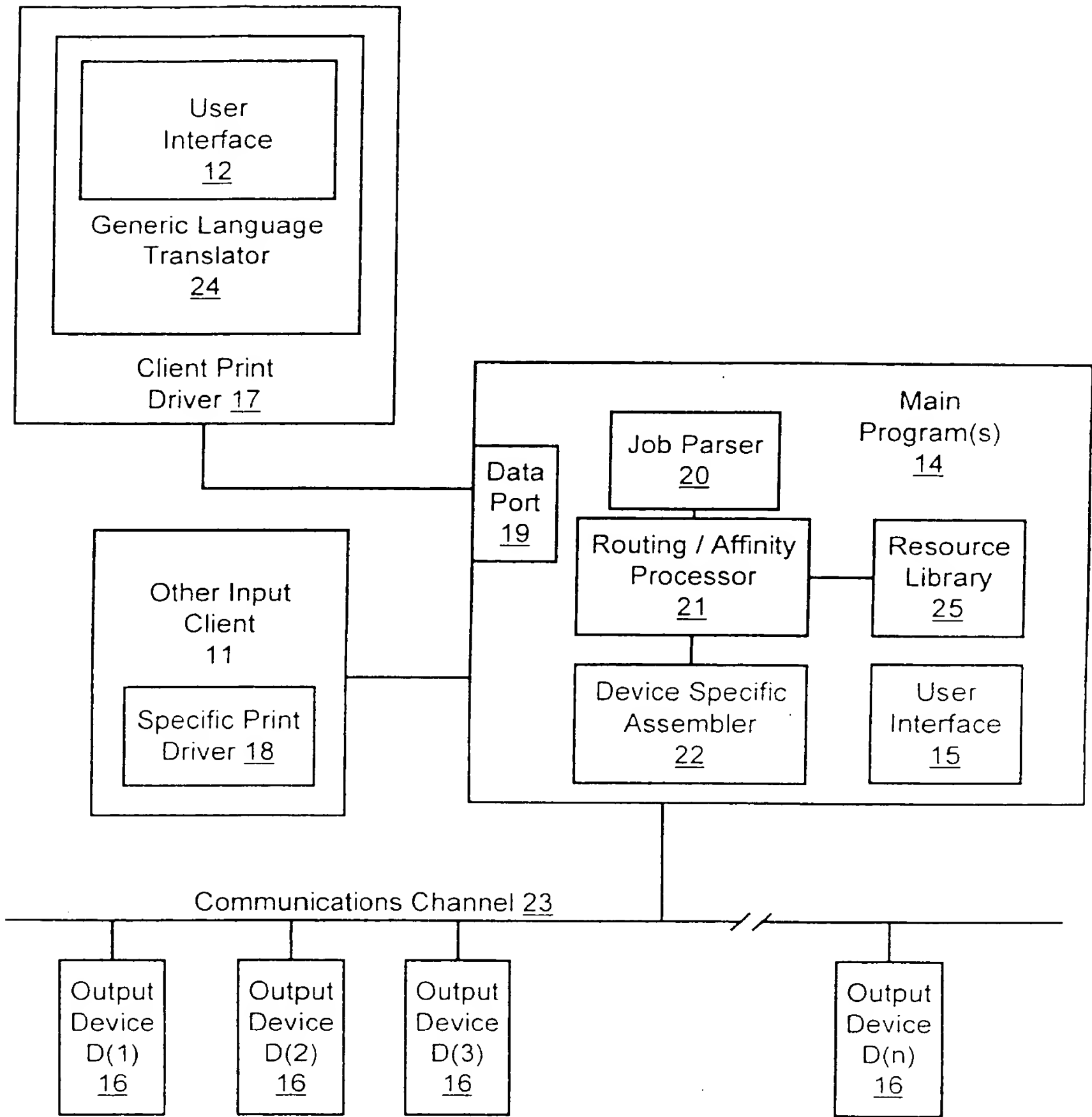


FIG.2

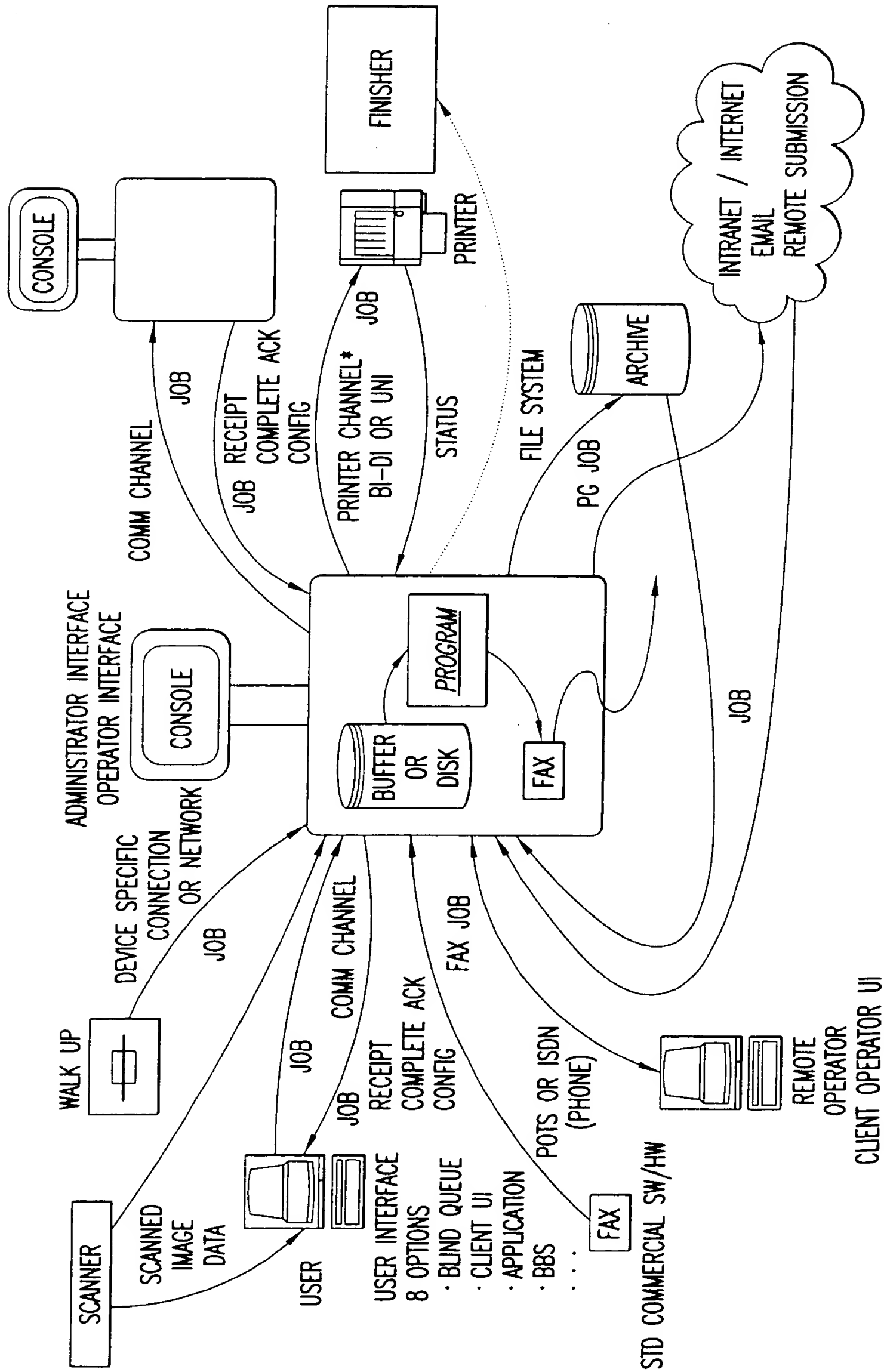


FIG.3

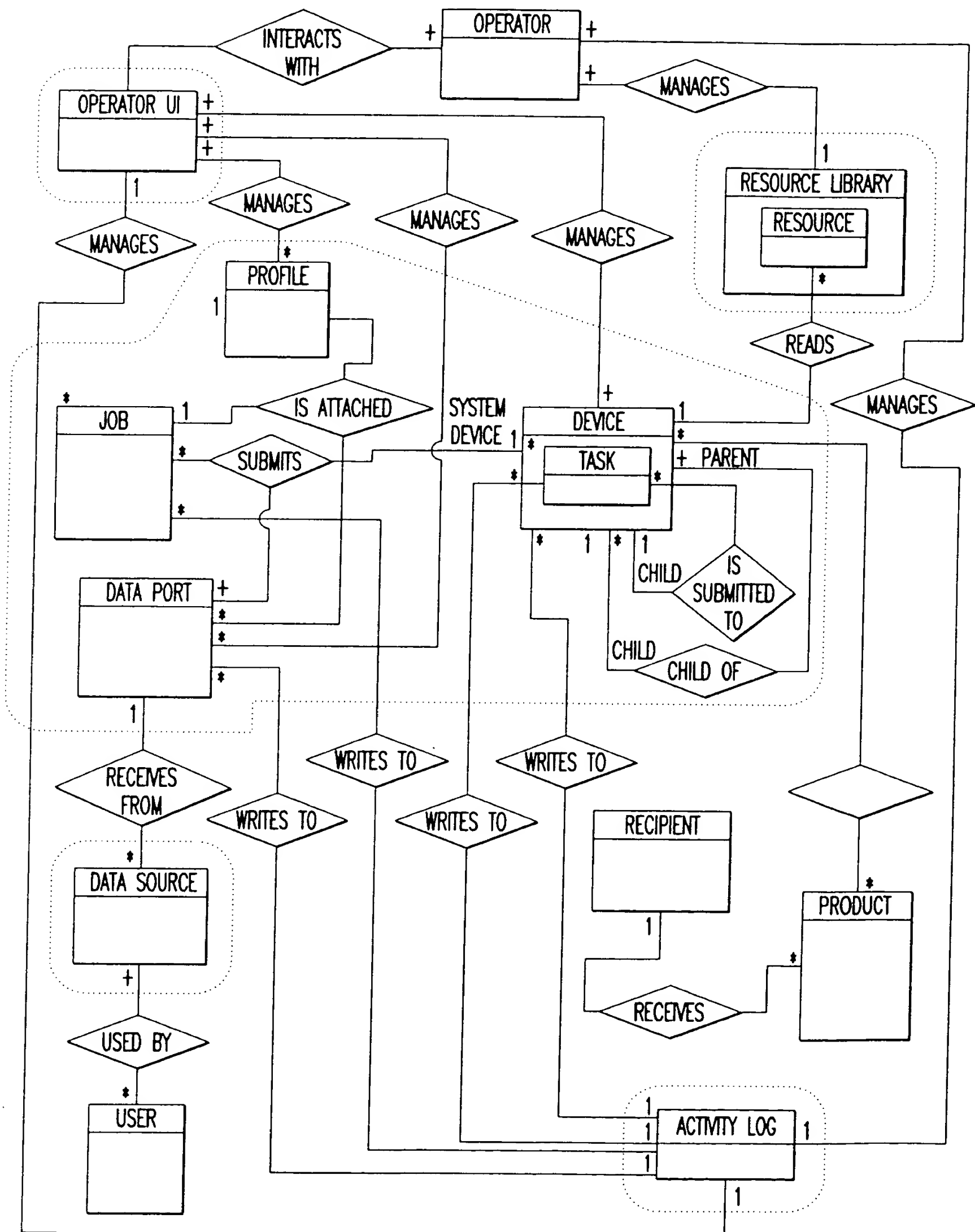


FIG.4

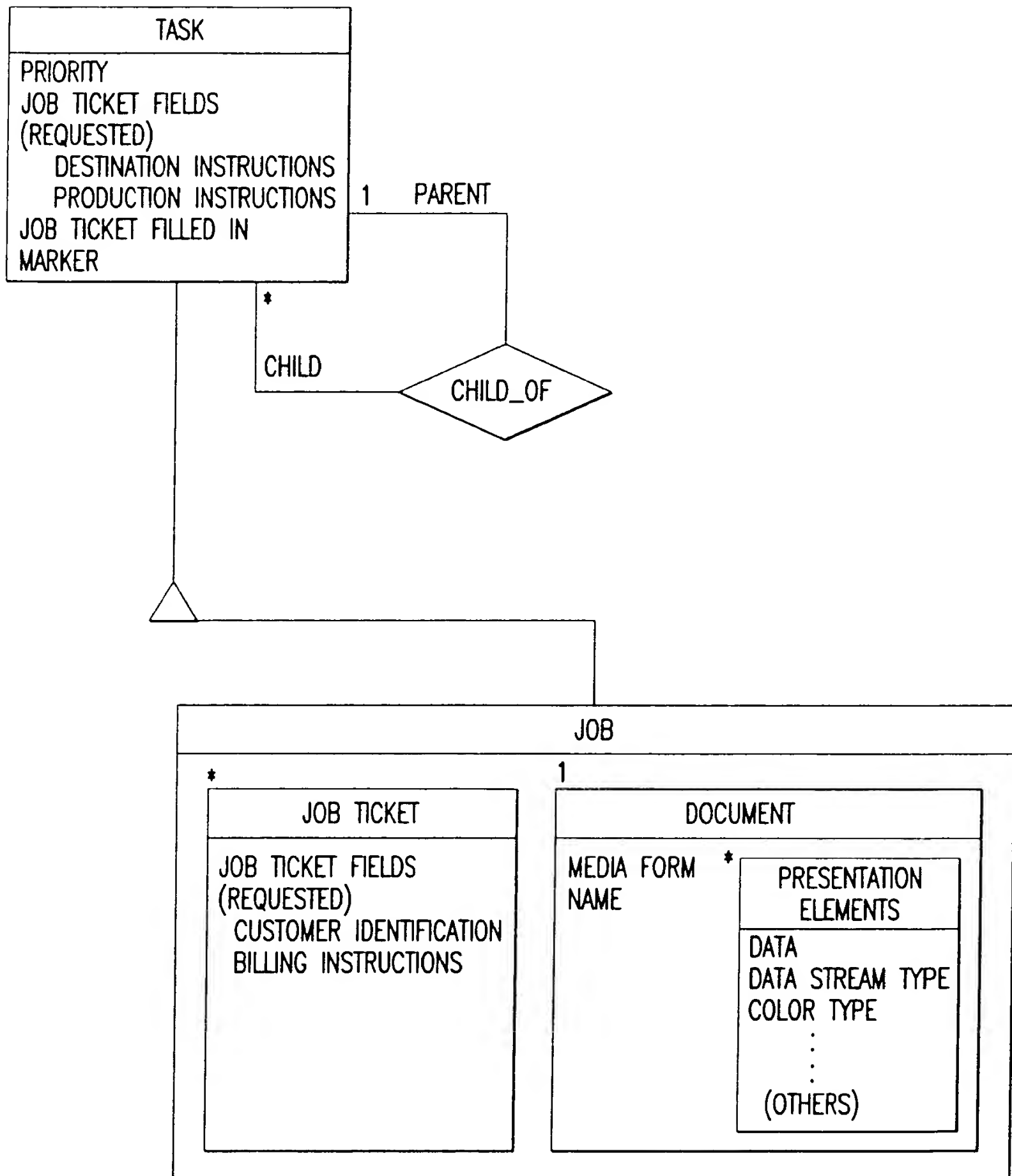


FIG.5

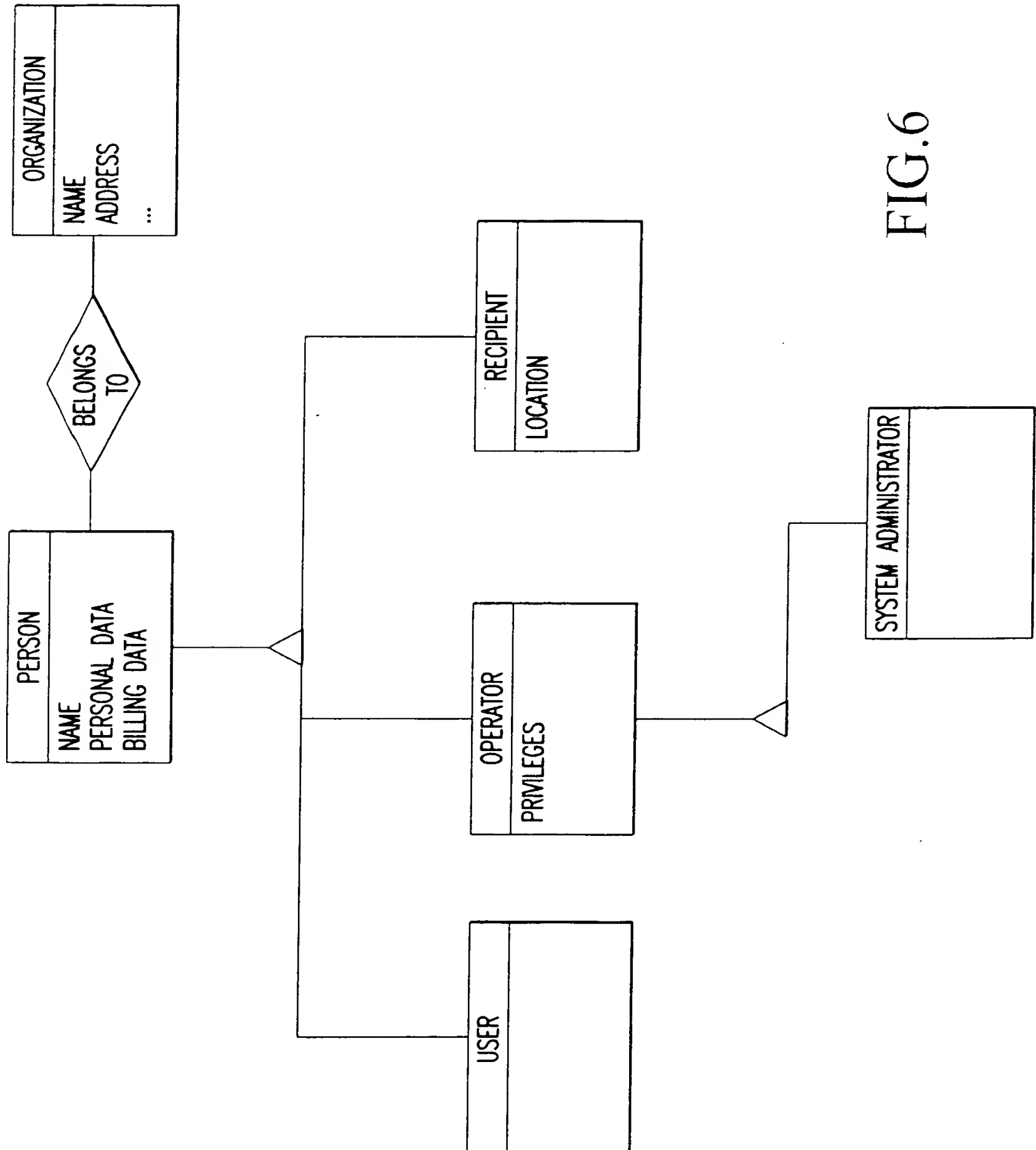


FIG.6

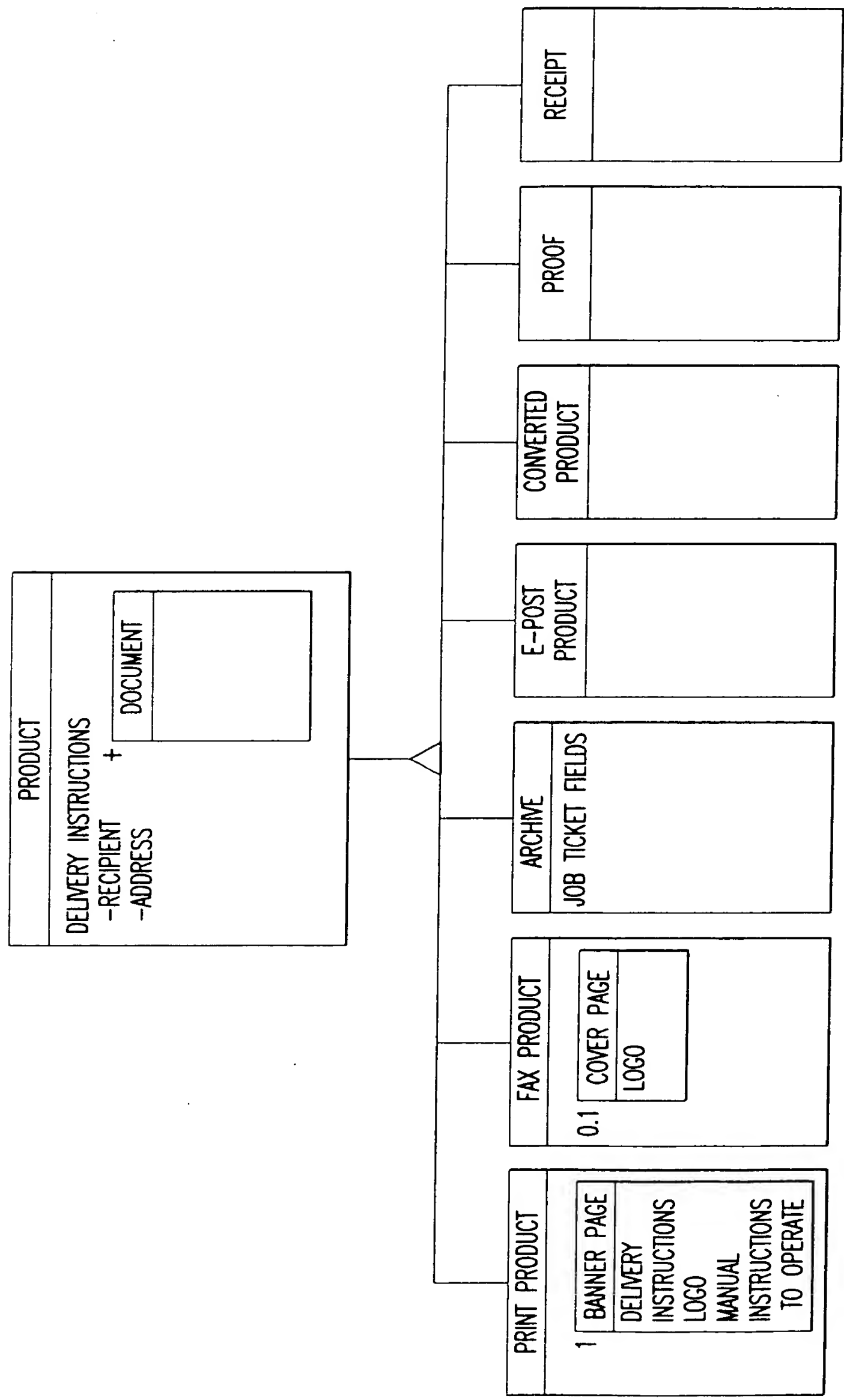


FIG.7

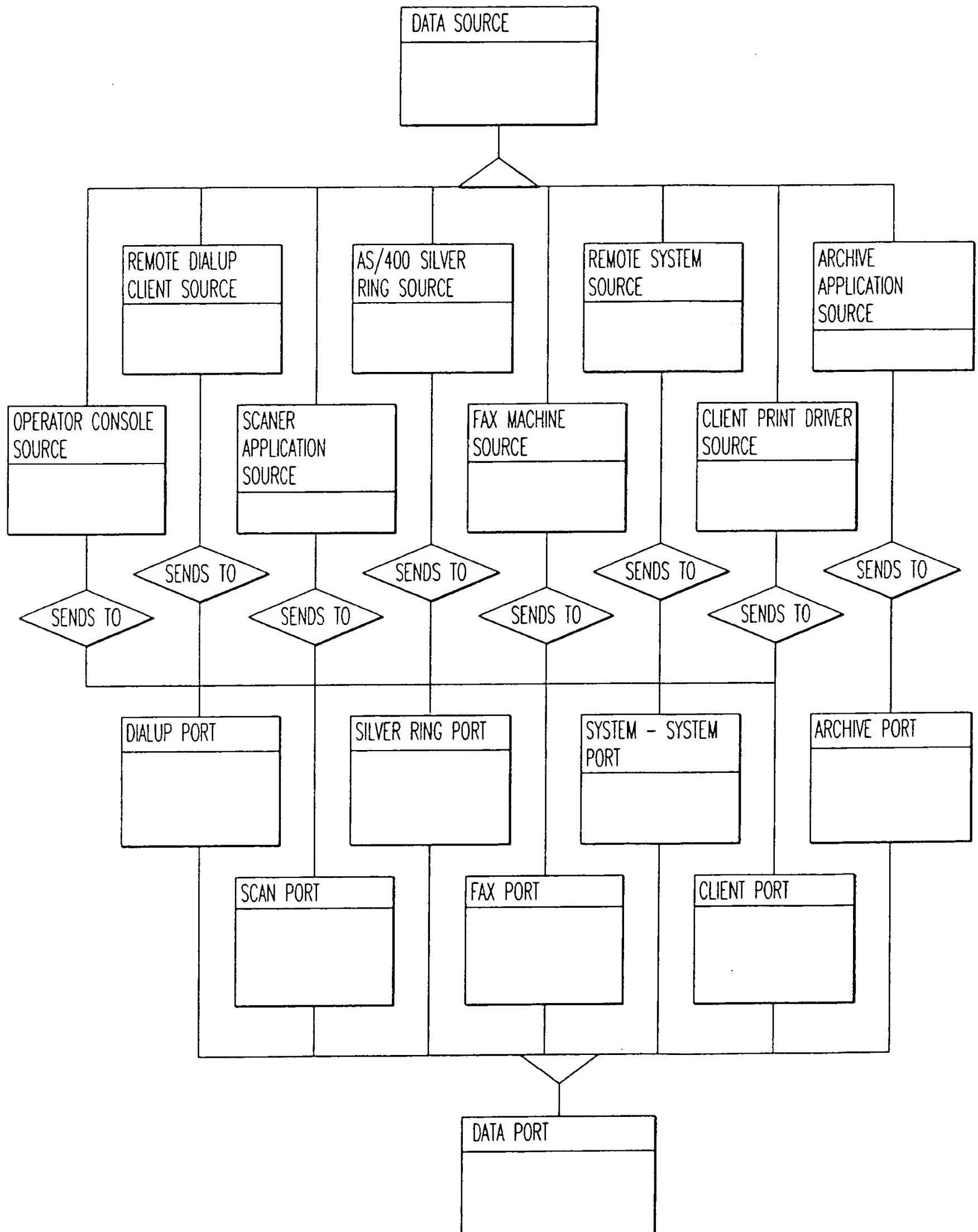


FIG.8

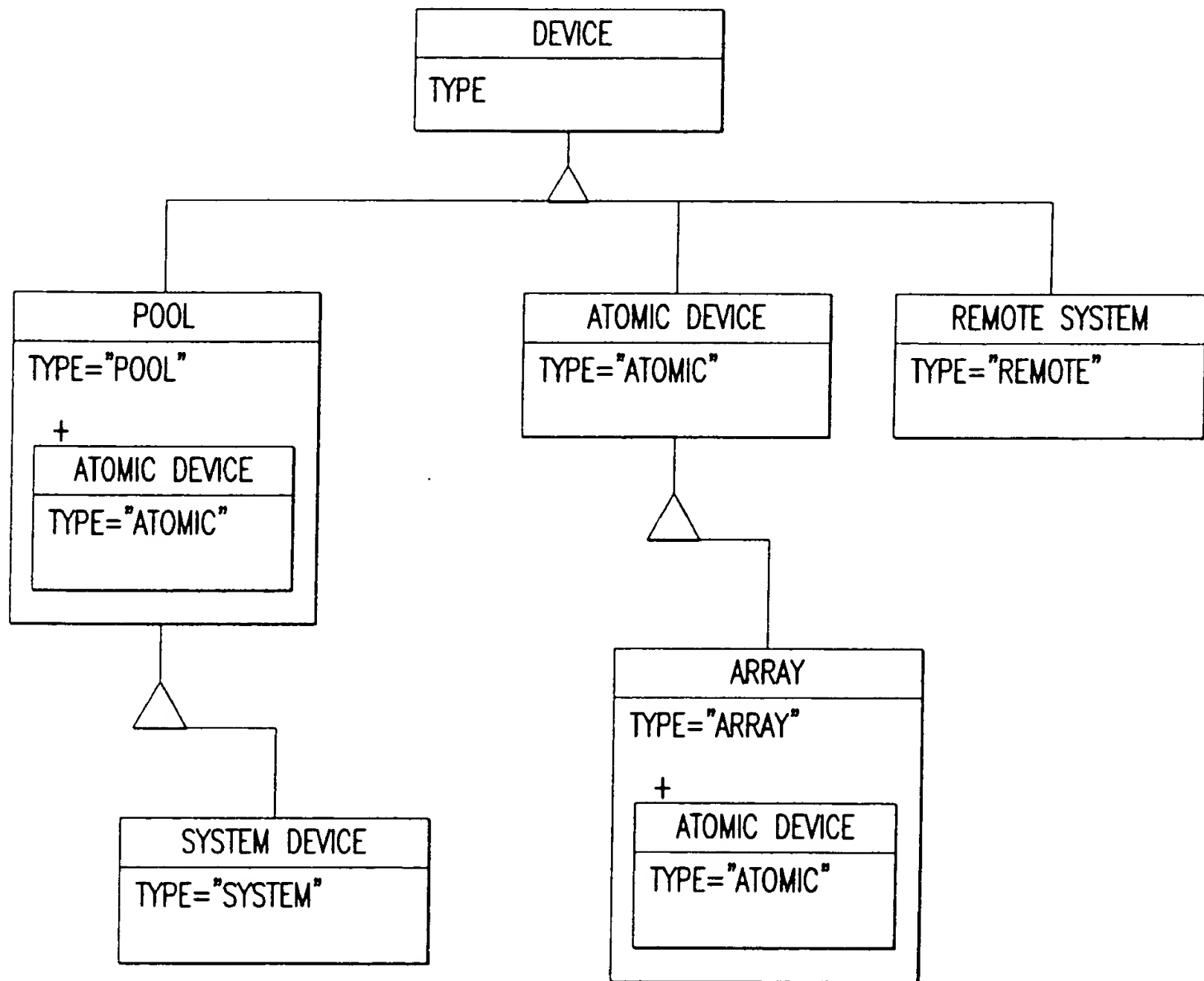


FIG.9

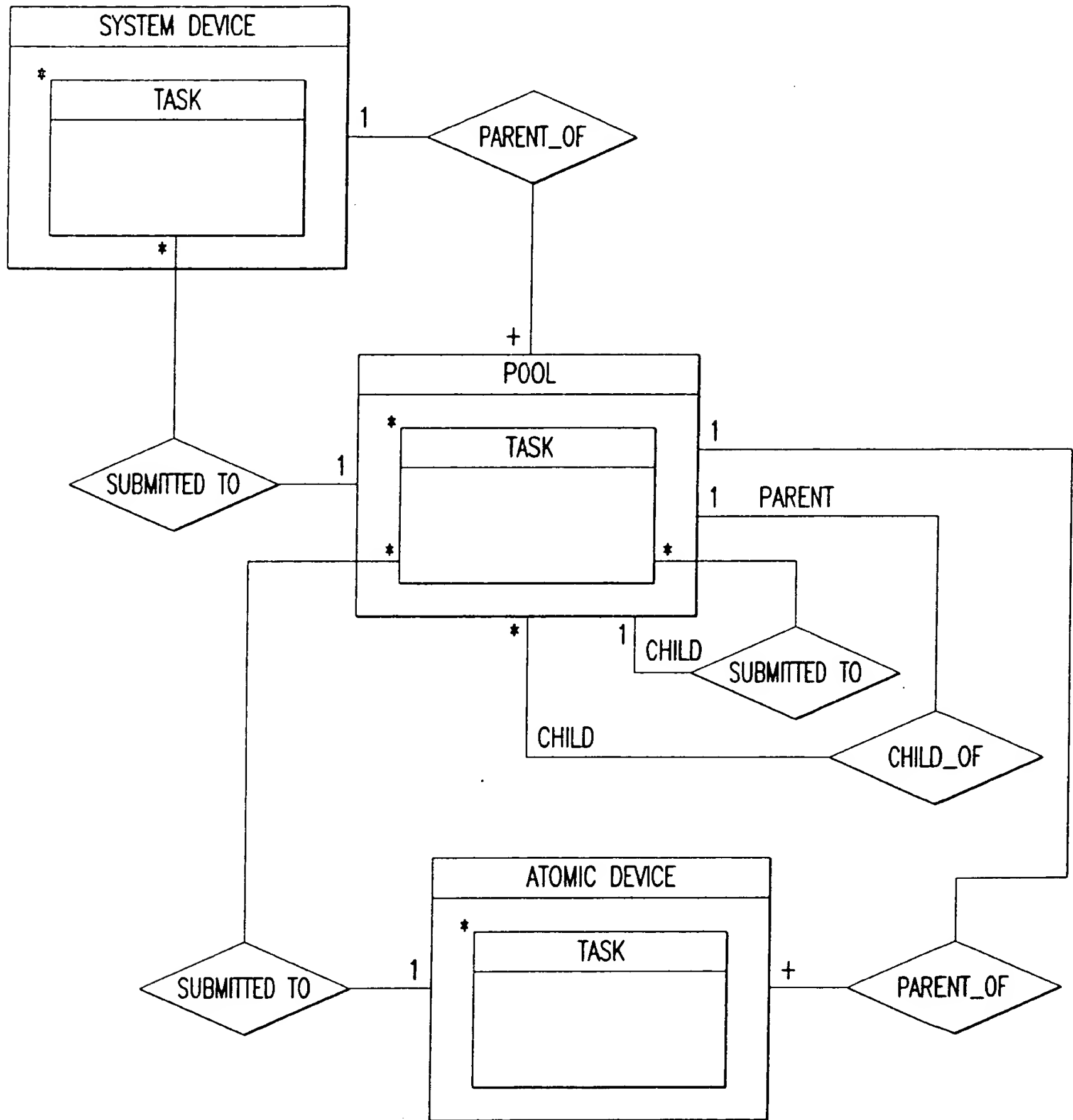


FIG.10

```

life-cycle  PrintGate:  (
                ( startup . #device_log_record ) .
                ( Production~ || Operation* || ResourceDelivery~ ) .
                ( shutdown . #device_log_record )
            )*

Production  =      submit_job .
                ( #needs_attention . #job_log_record )* .
                ( #needs_resource . #job_log_record )* .
                ( #proof . #job_log_record )* .
                ( #product . #receipt . #job_log_record . #task_log_record+ )

Operation   =      Administration | ControlPrintGate

ResourceDelivery  =      ( deliver_resource | deliver_resource_unavailable ) .
                        #resource_log_record

Administration    =      DeviceMgmt | PortMgmt
                        ControlPrintGate =      ( enter_job_password .
                        #job_log_record ) |
                        ( generate_encryption_key_pair . #generate_key_log_record ) |
                        JobMgmt |
                        TaskMgmt |
                        ProfileMgmt |
                        WorkQuery

DeviceMgmt =      (
                    (
                        add_device |
                        move_device |
                        copy_device |
                        change_device |
                        remove_device |
                        hold_device |
                        release_device |
                        ( query_device_configuration . #display_device_configuration )
                    ) .
                    #device_log_record
                ) |
                WorkQuery

```

FIG. 11

```
PortMgmt    =    (
                add_port |
                change_port |
                delete_port |
                hold_port |
                release_port |
                ( query_port . #port_properties )
            ) .
            #port_log_record

JobMgmt      =    (
                change_job_information |
                delete_job |
                ( view_job_document . #document_view ) |
                ( query_job_information . #job_information )
            ) .
            #job_log_record

TaskMgmt     =    ( add_task |
                change_task |
                cancel_task |
                hold_task |
                release_task |
                ( query_task . #display_task_properties ) ) .
            #task_log_record

ProfileMgmt  =    (
                create_profile |
                change_profile |
                delete_profile |
                ( query_profile . #display_profile_properties )
            ) .
            #profile_log_record

WorkQuery    =    ( query_device_status . #display_device_status .
            #device_log_record )
```

FIG.12

Operation Schema Listings

Operation Schema: Add Device

Operation: **add_device**

Description: Adds a new device to the PrintGate system.

Reads: **supplied name:** device_name,
 supplied privilege
 supplied new_parent : device_name
 parent_id: new_parent.id

Changes: **new new_device :** device,
 new new_link : is_child_of

Sends: activity_log:{device_log_record}

Assumes:

Result: If privilege allows, and
 if new_parent.type is not atomic, and
 if new_device.type will not be 'system'
 then
 new_device.name has been initialized to device_name.
 new_device.id has been set to default value.
 new_device.characteristics and new_device.priority have been
set to default values,
 new_link.parent has been set to new_parent.id,

FIG.13

App No.: 09/175,905 Docket No.: 0142-0317P
Inventor: Dennis W. HICKS et al.
Title: AUTOMATIC NETWORK DEVICE SELECTION AND
DOCUMENT DELIVERY SYSTEM
REPLACEMENT SHEET

new_link.child has been set to new_device.id.

device_log_record has been sent to the activity_log

FIG.14

Operation Schema: Add Port

Operation: add_port

Description: Operator adds a Data Port to the system.

Reads: supplied port_name,
 supplied port_configuration,
 supplied privileges.

Changes: new Data_Port

Sends: Activity Log:{port_log_record}

Assumes:

Result: If port_name does not refer to any existing ports, and privileges
allows this

operation then,

A new Data_Port has been added to the system.

Data_Port.name has been set to port_name.

Data_Port.configuration has been set to port_configuration.

Data_Port.held has been cleared.

port_log_record has been sent to the Activity Log

FIG.15

Operation Schema: Add Task

Operation: `add_task`

Description: Adds a new task for a job on to the system device.

Reads: `supplied task_instructions,`
 `supplied job_id,`
 `supplied privileges,`
 `job with job.id = job_id.`

Changes: `new task,`
 `device with device.type = "system".`

Sends: `activity_log: {task_log_record}`

Assumes:

Result: If privileges allow this operation and
 `job_id` identifies a job with `job.id = job_id` then
 A new Task has been added to the system Device,
 Task.`job_id` has been set to `job_id`,
 Task.instructions have been set to `task_instructions`.

`task_log_record` has been sent to the `activity_log`.

FIG.16

Operation Schema: Cancel Task

Operation: `cancel_task`

Description: Removes a job's task from the system

Reads: `supplied task_id`
 `supplied privileges`

Changes: `delete Task with task.id = task_id`
 `device with device_id = task.device_id`

Sends: activity log: {`task_log_record`}

Assumes:

Result: If privileges allows this operation then
 task has been removed from device.

`task_log_record` has been sent to the activity log.

FIG.17

Operation Schema: Change Device

Operation: **change_device**

Description: Changes a device's name, characteristics and priority.

Reads: **supplied** old_name : device_name
 supplied new_name: device_name,
 supplied new_characteristics : characteristics,
 supplied new_priority : priority,
 supplied privilege
 supplied dev_id : device_id

Changes: changed_device : device with device.name=old_name and
device.id=dev_id

Sends: activity_log:{device_log_record}

Assumes:

Result: If privilege allows and
 if changed_device.hold is set, then
 changed_device.name has been set to new_name.
 changed_device.characteristics has been set to
new_characteristics.
 changed_device.priority has been set to new_priority.

device_log_record has been sent to the activity_log

FIG.18

Operation Schema: Change Job Information

Operation: **change_job_information**

Description: Operator changes customer and billing information for a given job.

Reads: **supplied job_identification,**
 supplied new_billing_information,
 supplied new_customer_information
 supplied privileges.

Changes: Job with Job_id = job_identification

Sends: Activity Log: {job_record(s)}

Assumes: The new billing and customer information is valid.

Result: If privileges allows operation then,
 Job.billing_information has been set to new_billing_information.
 Job.customer_information has been set to
new_customer_information.

job_record(s) has been sent to the Activity Log.

FIG.19

Operation Schema: Change Port

Operation: change_port

Description: Operator changes the configuration of a Data Port

Reads: supplied old_port_name,
 supplied new_port_name,
 supplied new_configuration,
 supplied privileges.

Changes: Initial Data_Port with Data_Port.name = old_port_name.

Sends: Activity Log:{port_log_record}

Assumes:

Result: If new_port_name does not refer to any existing ports,
privileges allows operation, and

 Data_Port.held is set then,

 Final Data_Port.name has been set to new_port_name.

 Final Data_Port.configuration has been set to

new_configuration.

 port_log_record has been sent to the Activity Log

FIG.20

Operation Schema: Change Profile

Operation: `change_profile`

Description: Changes the attributes of an existing customer/user profile.

Reads: `supplied old_name,`
 `supplied new_name,`
 `supplied capabilities,`
 `supplied new_default_job_ticket,`
 `supplied privileges.`

Changes: `profile with profile.name = old_name`

Sends: `activity_log: {profile_log_record}`

Assumes:

Result: If `new_name` does not refer to any existing `profile.name` and
privileges allows

 this operation then

`Profile.name` is set to `new_name`,

`Profile.capabilities` is set to `new_capabilities`,

`Profile.default_job_ticket` is set to `new_default_job_ticket`.

`profile_log_record` has been sent to the `activity_log`.

FIG.21

Operation Schema: Change Task

Operation: change_task

Description: Changes a task's attributes or device assignment

Reads: supplied task_id,
supplied new_instructions,
supplied privileges.

Changes: Task with Task.id = task_id,
original_device: *Initial* device with device.id = task.device_id,
system_device: device with device.type = "system".

Sends: activity_log: {task_log_record}

Assumes:

Result: If privileges allows this operation then
Task.instructions has been set to new_instructions.
Task.hold_status has been set on. *
If *Final* Task.instructions can not be performed on
original_device then
Task has been removed from original_device,

FIG.22

App No.: 09/175,905

Docket No.: 0142-0317P

Inventor: Dennis W. HICKS et al.

Title: AUTOMATIC NETWORK DEVICE SELECTION AND
DOCUMENT DELIVERY SYSTEM
REPLACEMENT SHEET

Task has been assigned to system_device.

task_log_record has been sent to the activity_log.

* Note: this means that task will need to be released later.

FIG.23

Operation Schema: Copy Device

Operation: copy_device

Description: Copies a device to a different pool.

Reads: supplied new_parent : device_id
 supplied child_id : device_id
 supplied privilege
 parent : device with device.id = new_parent
 child : device with device.id = child_id

Changes: new new_link : is_child_of

Sends: activity_log:{device_log_record}

Assumes:

Result: If privilege allows, and
 if child has not become an ancestor of child, and
 if parent.type is not atomic, and
 if child.type is not system,
 then
 new_link.parent has been set to new_parent and
 new_link.child has been set to child_id.

FIG.24

App No.: 09/175,905

Docket No.: 0142-0317P

Inventor: Dennis W. HICKS et al.

Title: AUTOMATIC NETWORK DEVICE SELECTION AND
DOCUMENT DELIVERY SYSTEM
REPLACEMENT SHEET

device_log_record has been sent to the activity_log

FIG.25

Operation Schema: Create Profile

Operation: create_profile

Description: Adds a new customer/user profile to the system.

Reads: supplied name,
 supplied privileges.

Changes: new profile

Sends: activity_log: {profile_log_record}

Assumes:

Result: If name does not refer to any existing profile.name and
privileges allows this

 operation then

 A new profile has been added to the system,

 Profile.name has been set to name,

 Profile.capabilities has been set to profile.capabilities from

profile with

 profile with profile.name = "Default".

 profile_log_record has been sent to the activity_log.

FIG.26

Operation Schema: Delete Job

Operation: delete_job

Description: Operator deletes a Job and all of its associated Tasks from the system.

Reads: supplied job_identification,
supplied privileges.

Changes: delete Job with Job.id = job_identification,
delete all Tasks with Task.job_id = job_identification,
for each Task above,
Device with Device.id = Task.device_id.

Sends: Activity Log: {job_log_record}

Assumes:

Result: If privileges allows operation then,
Job has been removed from system.
All Tasks associated with Job have been removed from their
respective
devices.

job_log_record has been sent to Activity Log.

FIG.27

Operation Schema: Delete Port

Operation: delete_port

Description: Operator removes a Data Port from the system.

Reads: supplied port_name,
supplied privileges.

Changes: delete Data_Port with Data_Port.name = port_name.

Sends: Activity Log:{port_log_record}

Assumes:

Result: If privileges allows the operation, and Data_Port.held is set
then,

The Data_Port with Data_Port.name = port_name has been removed
from
the system.

port_log_record has been sent to the Activity Log.

FIG.28

Operation Schema: Delete Profile

Operation: `delete_profile`

Description: Remove an existing customer/user profile from the system.

Reads: `supplied name,`
`supplied privileges.`

Changes: `delete profile with profile.name = name.`

Sends: `activity_log: {profile_log_record}`

Assumes:

Result: `If privileges allows this operation then`
`named profile is removed from the system.`

`profile_log_record` has been sent to the `activity_log`.

FIG.29

Operation Schema: Deliver Resource

Operation: deliver_resource

Description: Resource Library delivers a Resource to a Device that is processing a Task.

Reads: supplied device_name,
supplied resource_name,
supplied resource_type,
supplied resource_data.

Changes: Device with Device.name = device_name.

Sends: Activity Log:{job_log_record}

Assumes:

Result: If Device.processes.resource_name is resource_name,
Device.processes.resource_type is resource_type, and
Device.processes.resource_data is NULL then,
Device.processes.resource_data has been set to
resource_data.

job_log_record has been sent to the Activity Log.

FIG.30

Operation Schema: Deliver Resource Unavailable

Operation: `deliver_resource_unavailable`

Description: Resource Library indicates that a Resource needed by a Device that is processing a Task
 cannot be delivered.

Reads: `supplied device_name,`
 `supplied resource_name,`
 `supplied resource_type,`
 Device with Device.name = device_name.

Changes: Task with Taks.id = Device.task_id.

Sends: Activity Log:{job_log_record}
 Operator:{attention_required}

Assumes:

Result: If Device.processes.resource_name is set to resource_name,
 Device.processes.resource_type is set to resource_type, and
 Device.processes.resource_data is set to NULL then,
 Task.held has been set.
 Task.intervention_required has been set.
 attention_required has been sent to the Operator.

job_log_record has been sent to the Activity Log

FIG.31

Operation Schema: Enter Job Password

Operation: `enter_job_password`

Description: Operator enters password to allow Job to be processed.

Reads: `supplied job_identification,`
 `supplied password,`
 `supplied privileges.`

Changes: Job with Job.id = `job_identification`

Sends: Activity Log: {`job_log_record`}

Assumes:

Result: If operator_privileges allows operation, and
 Job.password_required is set, and
 Job.password is set to password then,
 Job.password_required is cleared.

`job_log_record` has been sent to Activity Log.

FIG.32

Operation Schema: Generate Encryption Key Pair

Operation: generate_encryption_key_pair

Description: Operator generates a encryption/decryption key pair for job encryption.

Reads: supplied privileges,
supplied profile_name.

Changes: Profile with Profile.name = profile_name

Sends: Operator: {public_decryption_key},
Activity Log: {profile_log_record}.

Assumes:

Result: If privileges allows the operation then,
A encryption/decryption key pair has been generatred.
Profile.encryption_key has been set to the encryption key.
Profile.decryption_key has been set to the decryption key.
The decryption key has been sent to the Operator.

profile_log_record has been sent to the Activity Log

FIG.33

Operation Schema: Hold Device

Operation: `hold_device`

Description: Pauses a device's execution

Reads: `supplied dev_id: device_id`
 `supplied privilege`

Changes: `held_device : device with device.id equal to dev_id`

Sends: `activity_log:{device_log_record}`

Assumes:

Result: If privilege allows, and `device.hold_status` for `held_device` is
cleared, then

 The `device.hold_status` for `held_device` has been set.

`device_log_record` has been sent to the `activity_log`

FIG.34

Operation Schema: Hold Port

Operation: hold_port

Description: Operator holds a Data Port to prevent Job submission.

Reads: supplied port_name ,
supplied privileges.

Changes: Data_Port with Data_Port.name = port_name.

Sends: Activity Log:{port_log_record}

Assumes:

Result: If Data_Port.held is not set, and privileges allows operation
then,

Data_Port.held has been set.

port_log_record has been sent to the Activity Log.

FIG.35

Title: AUTOMATIC NETWORK DEVICE SELECTION AND
DOCUMENT DELIVERY SYSTEM
REPLACEMENT SHEET

Operation Schema: Hold Task

Operation: **hold_task**

Description: Places a task on hold so that it will not continue to be processed.

Reads: **supplied task_id,**
 supplied privileges.

Changes: task with task.id = task_id

Sends: activity_log: {task_log_record}

Assumes:

Result: If privileges allows this operation and
 if *Initial* task.hold_status is cleared then
 Final task.hold_status has been set.

task_log_record has been sent to the activity_log.

FIG.36

Operation Schema: Move Device

Operation: move_device

Description: Moves a device to a different pool.

Reads: supplied old_parent : device_id
 supplied new_parent : device_id
 supplied child_id : device_id
 supplied privilege
 parent : device with device.id = new_parent
 child : device with device.id = child_id

Changes: new new_link : is_child_of
 delete old_link : is_child_of with is_child_of.parent = old_parent and
 is_child_of.child = child_id

Sends: activity_log:{device_log_record}

Assumes:

Result: If privilege allows, and
 if child has not become an ancestor of child, and
 if new_parent.type is not atomic, and
 if child.type is not system, and
 if child.hold is set,
 then
 old_link has been removed and

FIG.37

new_link.parent has been set to new_parent and
new_link.child has been set to child_id.

device_log_record has been sent to the activity_log

FIG.38

Operation Schema: Query Device Configuration

Operation: `query_device_configuration`

Description: Reports the current configuration and properties of a device

Reads: `supplied device_name,`
`supplied privileges,`
`device with device.name = device_name.`

Changes:

Sends: `operator: {device_properties}`
`activity_log: {device_log_record}`

Assumes:

Result: `If privileges allow this operation then`
`device_properties sent to operator.`

`device_log_record has been sent to the activity_log.`

FIG.39

Operation Schema: Query Device Status

Operation: `query_device_status`

Description: Reports the current processing status of a system device

Reads: `supplied device_name,`
`supplied privileges,`
`device with device.name = device_name.`

Changes:

Sends: `operator: {device_status}`
`activity_log: {device_log_record}`

Assumes:

Result: `If privileges allow this operation then`
`device_status sent to operator.`

`device_log_record has been sent to the activity_log.`

FIG.40

Operation Schema: Query Job Information

Operation: **query_job_information**

Description: Reports the current job-level properties of a job

Reads: **supplied job_id,**
 supplied privileges,
 job with job.id = job_id.

Changes:

Sends: operator: {job_information}
 activity_log: {job_log_record}

Assumes:

Result: If privileges allow this operation then
 job_information sent to operator.

job_log_record has been sent to the activity_log.

FIG.41

Operation Schema: Query Port

Operation: query_port

Description: Reports the current properties and status of a data port

Reads: supplied port_name,
 supplied privileges,
 port with port.name = port_name.

Changes:

Sends: operator: {port_properties, port_status}
 activity_log: {port_log_record}

Assumes:

Result: If privileges allow this operation then
 port_properties and port_status sent to operator.

port_log_record has been sent to the activity_log.

FIG.42

Operation Schema: Query Profile

Operation: query_profile

Description: Reports the current properties of a profile

Reads: supplied profile_name,
 supplied privileges,
 profile with profile.name = profile_name.

Changes:

Sends: operator: {profile_properties}
 activity_log: {profile_log_record}

Assumes:

Result: If privileges allow this operation then
 profile_properties sent to operator.

profile_log_record has been sent to the activity_log.

FIG.43

Operation Schema: Query Task

Operation: query_task

Description: Reports the current properties and status of a task

Reads: supplied task_id,
 supplied privileges,
 Task with Task.id = task_id.

Changes:

Sends: Operator: {task_properties, task_status}
 activity_log: {task_log_record}

Assumes:

Result:

If privileges allow this operation then
 task_properties and task_status sent to Operator.

task_log_record has been sent to activity_log.

FIG.44

Operation Schema: Release Device

Operation: **release_device**

Description: Resumes a device's execution after it has been paused.

Reads: **supplied dev_id: device_id,**
 supplied privilege

Changes: **released_device : device with device.id equal to device_id**

Sends: **activity_log:{device_log_record}**

Assumes:

Result: If privilege allows, and device.hold for released_device is set,
then

 The device.hold for released_device has been cleared.

 device_log_record has been sent to the activity_log

FIG.45

Operation Schema: Release Port

Operation: release_port

Description: Operator releases a Data Port to allow Job submission;

Reads: supplied port_name,
supplied privileges.

Changes: Data_Port with Data_Port.name = port_name.

Sends: Activity Log:{port_log_record}

Assumes:

Result: If privileges allows operation then,
Data_Port.status has been set to Ready.

port_log_record has been sent to the Activity Log

FIG.46

Operation Schema: Release Task

Operation: `release_task`

Description: Resumes a task's operation so that it will continue to be processed.

Reads: `supplied task_id`,
 `supplied privileges`.

Changes: `task with task.id = task_id`

Sends: `activity_log: {task_log_record}`

Assumes:

Result: If privileges allows this operation and
 if *Initial* `task.hold_status` is set on then
 Final `task.hold_status` has been cleared.

`task_log_record` has been sent to the `activity_log`.

FIG.47

Operation Schema: Remove Device

Operation: remove_device

Description: Removes a device from a pool. If it is the only instance of that device in the system, the device is
removed from the PrintGate system.

Reads: supplied dev_id: device_id,
 supplied privilege
 supplied parent_id : device_id

Changes: delete link: is_child_of with is_child_of.child = dev_id and
is_child_of.parent = parent_id
 delete last_device : device with device.id = dev_id

Sends: activity_log:{device_log_record}

Assumes:

Result: If privilege allows and
 if last_device.hold is set, then
 link has been removed and
 If no other is_child_of with is_child_of.child = dev_id then
 last_device has been removed.
 Otherwise, last_device has not been removed.

 device_log_record has been sent to the activity_log

FIG.48

Operation Schema: Shutdown

Operation: shutdown

Description: Shuts down the PrintGate system.

Reads: supplied privilege

Changes: system_device : device with device.type equal to 'system'
all_devices : all devices in the system which have device.hold cleared

Sends:

Assumes:

Result: If privilege allows, and
if device.availability for system_device has been set then
system_device has had device.availability cleared
all_devices has had device.hold set

device_log_record has been sent to the activity_log.

FIG.49

Operation Schema: Startup

Operation: startup

Description: Starts the PrintGate system.

Reads: supplied privilege

Changes: system_device : device with device.type equal to 'system'
all_devices : all devices in the system which have device.hold set

Sends: activity_log: {device_log_record}

Assumes:

Result: If privilege allows, and
if device.availability for system_device has not been set then
system_device has had device.availability set.
all_devices has had device.hold cleared
Otherwise, no change occurs.

device_log_record has been sent to the activity_log.

FIG.50

Operation Schema: Submit Job

Operation: submit_job

Description: Data Source adds a Job to the system.

Reads: **supplied** port_name,
 supplied originator,
 supplied job_ticket,
 supplied document_data,
 Data_Port with Data_Port.name = port_name,
 Profile with Profile.name = originator,
 Profile with Profile.name = port_name,
 Profile with Profile.name = "Default".

Changes: new Job.

Sends: Product: {proof,product,receipt},
 Operator:{processing_job, completed_job, needs_attention},
 Resource Library:{resource_request},
 Activity Log:{job_log_record(s)}.

Assumes: document_data is in a supported PDL.

Result: If Data_Port.held is cleared then,
 A new job has been added to the system.
 Job.instructions, Job.customer_information and
Job.billing_information

FIG.51

have been set based on the job_ticket.
Job.document.data has been set to document_data.
Job.originator has been set to originator.
If Profile with Profile.name = Job.originator exists then,
 If Job.is_encrypted is set,
 Job.instructions, Job.customer_information,

 Job.billing_information and Job.document_data has

 been decrypted using Profile.decryption_key from Profile
with Profile.name
 Job.originator.
 Job.instructions, Job.customer_information and
 Job.billing_information has been merged with
Profile.default_instructions,
 Profile.default_customer_information and
 Profile.default_billing_information respectively from Profile with
Profile.name = Job.originator.
 Job.priority has been set to Profile.default_priority from Profile
with
 Profile.name = Job.originator.
Otherwise if Profile with Profile.name = Job.port exists then,
 If Job.is_encrypted is set,
 Job.instructions, Job.customer_information,
Job.billing_information and Job.document_data has been decrypted using
Profile.decryption_key from Profile with Profile.name = Job.port.
 Job.instructions, Job.customer_information and
 Job.billing_information has been merged with
 Profile.default_instructions,
Operation Schema: Submit Job (Continued)

FIG.52

Profile.default_customer_information and
Profile.default_billing_information respectively from Profile with Profile.name
=
Job.port.
Job.priority has been set to Profile.default_priority from Profile
with
Profile.name = Job.port.
Otherwise,
If Job.is_encrypted is set,
Job.instructions, Job.customer_information,
Job.billing_information and Job.document_data has been decrypted using
Profile.decryption_key from Profile with Profile.name = "Default".
Job.instructions, Job.customer_information and
Job.billing_information has been merged with
Profile.default_instructions,
Profile.default_customer_information and
Profile.default_billing_information respectively from Profile
with Profile.name = "Default".
Job.priority has been set to Profile.default_priority from Profile
with
Profile.name = "Default".
job_log_record(s) has been sent to the Activity Log.
processing_job indicator (i.e. lights) has been signaled to the Operator.
If operator attention was required then,
needs_intervention indicator (i.e. lights) has been signaled to
the Operator.
If external resources were required then,
resource_request has been sent to the Resource Library.
Job.status has been set to Done.

FIG.53

completed_job indicator (i.e. lights) has been signaled to the Operator.

If Job.instructions indicate that a proof was required then,

A proof has been generated.

The product has been generated.

A receipt has been generated.

Otherwise,

There was no effect on the system.

FIG.54

Operation Schema: View Job Document

Operation: view_job_document

Description: Operator views a representation of the document data of a Job.

Reads: supplied job_identification,
 supplied privileges,
 Job with Job.id = job_identification.

Changes:

Sends: Operator: {document_representation},
 Activity Log: {job_log_record}.

Assumes:

Result: If the privileges allows the operation then,
 Job.document_data has been converted into a viewable format.
 The viewable data has been presented to the Operator.

job_log_record has been sent to the Activity Log

FIG.55

FUSION NOTATION SUMMARY

OBJECT MODEL NOTATION

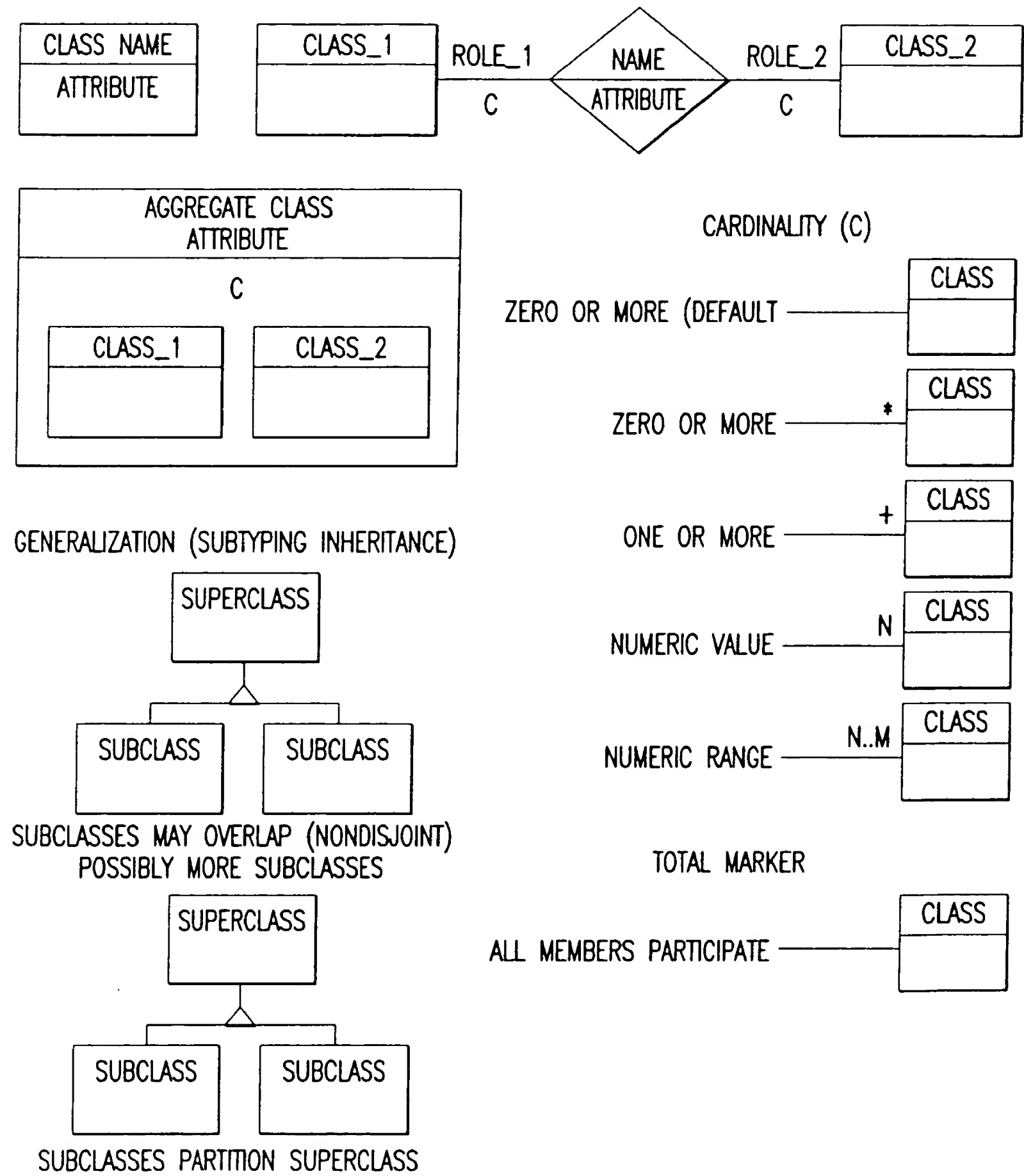


FIG.56

Life-Cycle Model Notation

life cycle [<i>Name</i> :] <i>Regular_Expression</i> (<i>LocalName</i> = <i>Regular_Expression</i>)*		
<i>Regular_Expressions:</i> event	Name	Any event name (operation) ,. Local name, or output event
	Concatenation	<i>x.y</i>
	Alternation	<i>x y</i>
	Repetition	
	Zero or more	<i>x*</i>
	One or more	<i>x+</i>
	Optional	[<i>x</i>]
	Interleaving	
	Grouping	(<i>x</i>)

Operation Model Notation

Operation:	operation identifier
Description:	<text> Description of operation
Reads:	<supplied values> <state components>
Changes:	<supplied values> <state components>
Sends:	<agent communication> <state components>
Assumes:	<assertions> (preconditions)
Result:	<assertions> (preconditions)

FIG.57

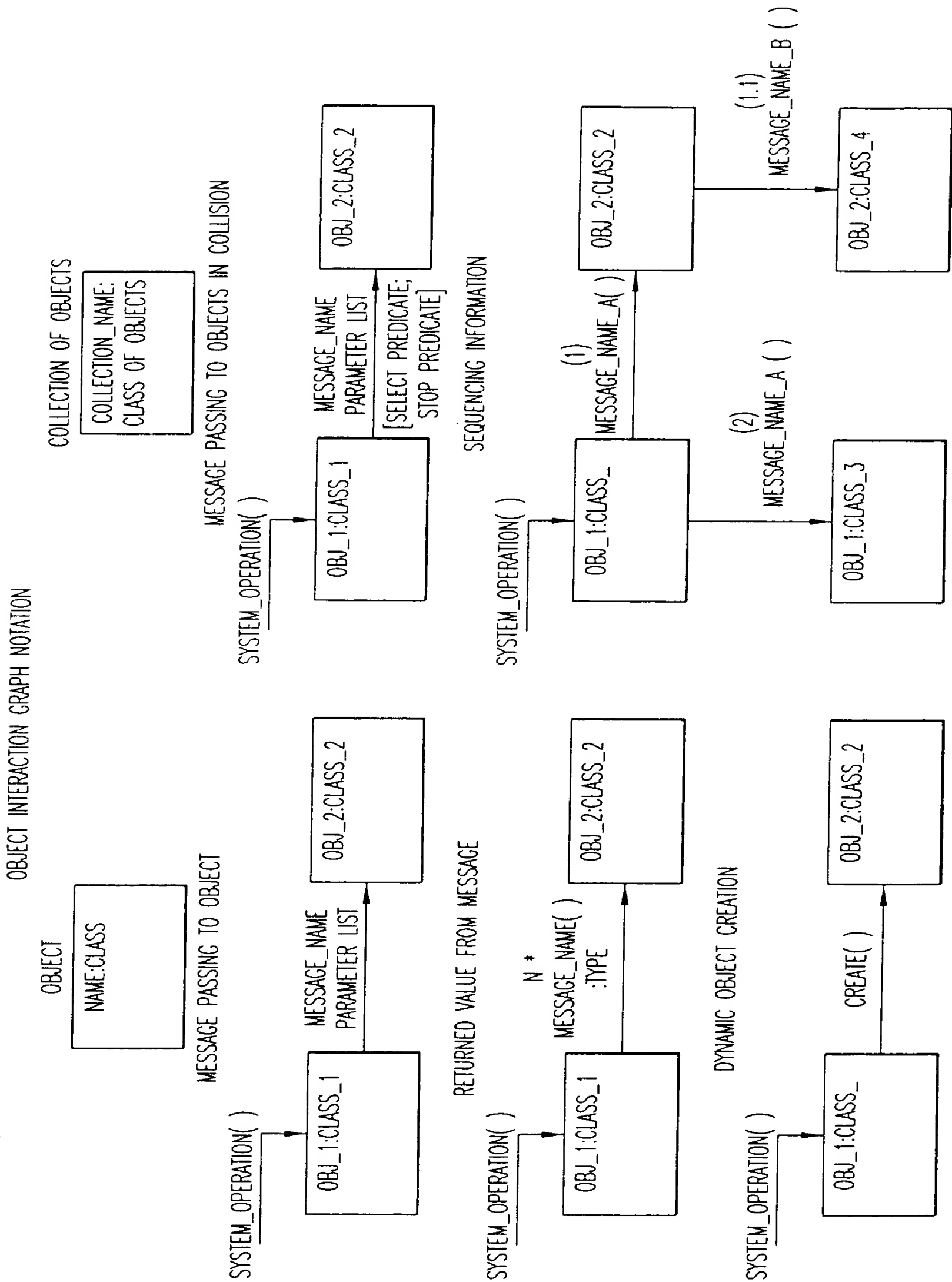


FIG.58

VISIBILITY GRAPH NOTATION

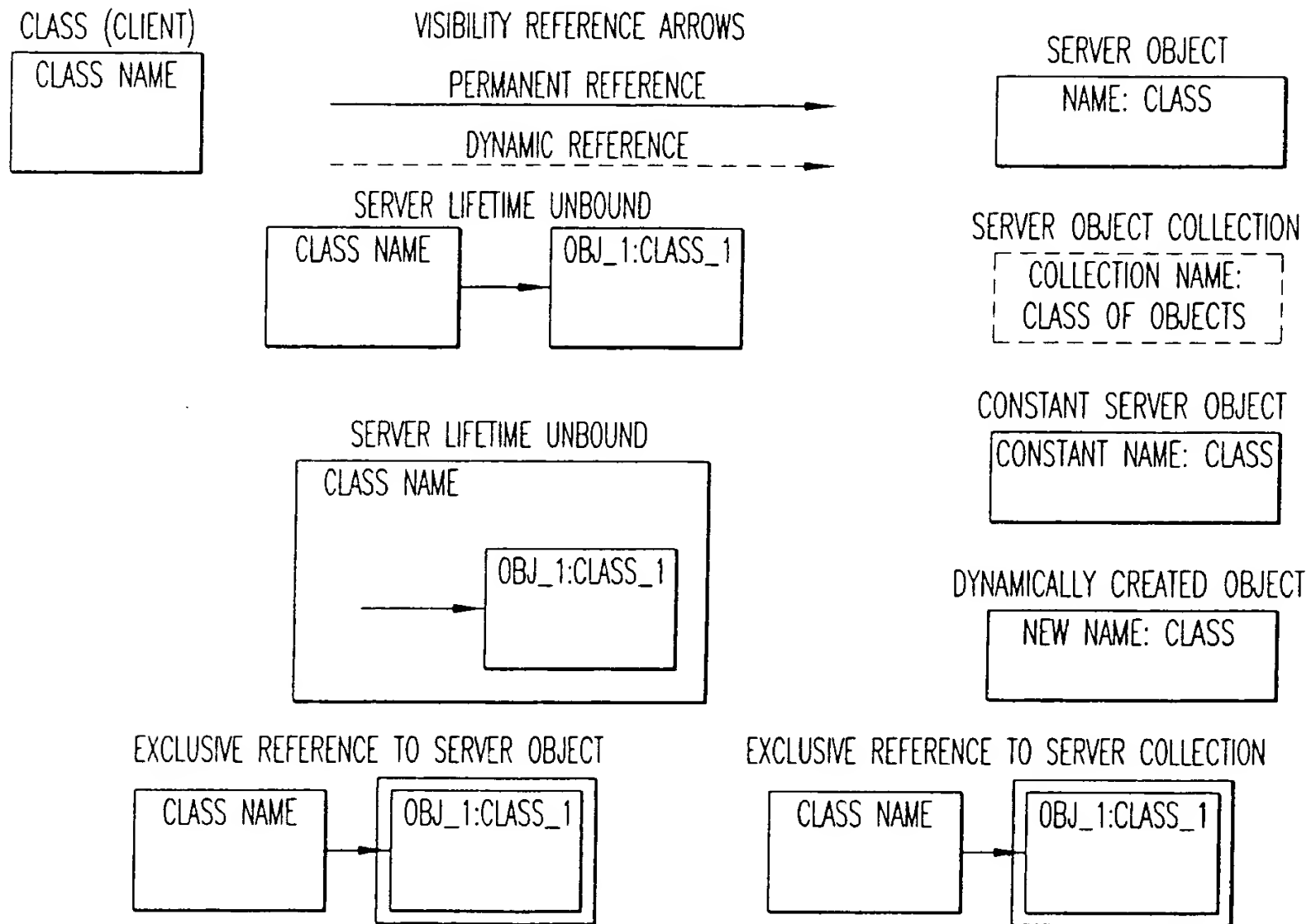


FIG.59

Class Description Notation:

```
class <ClassName> [isa < SuperClassNames>]
  // for each attribute
    [attribute] [mutability]<a_name> :[sharing][Binding}<Type>
    .
    .
    .
  // for each method
    [method] <m_name> <arglist>[:<Type>]
    .
    .
    .
endclass
```

FIG.60

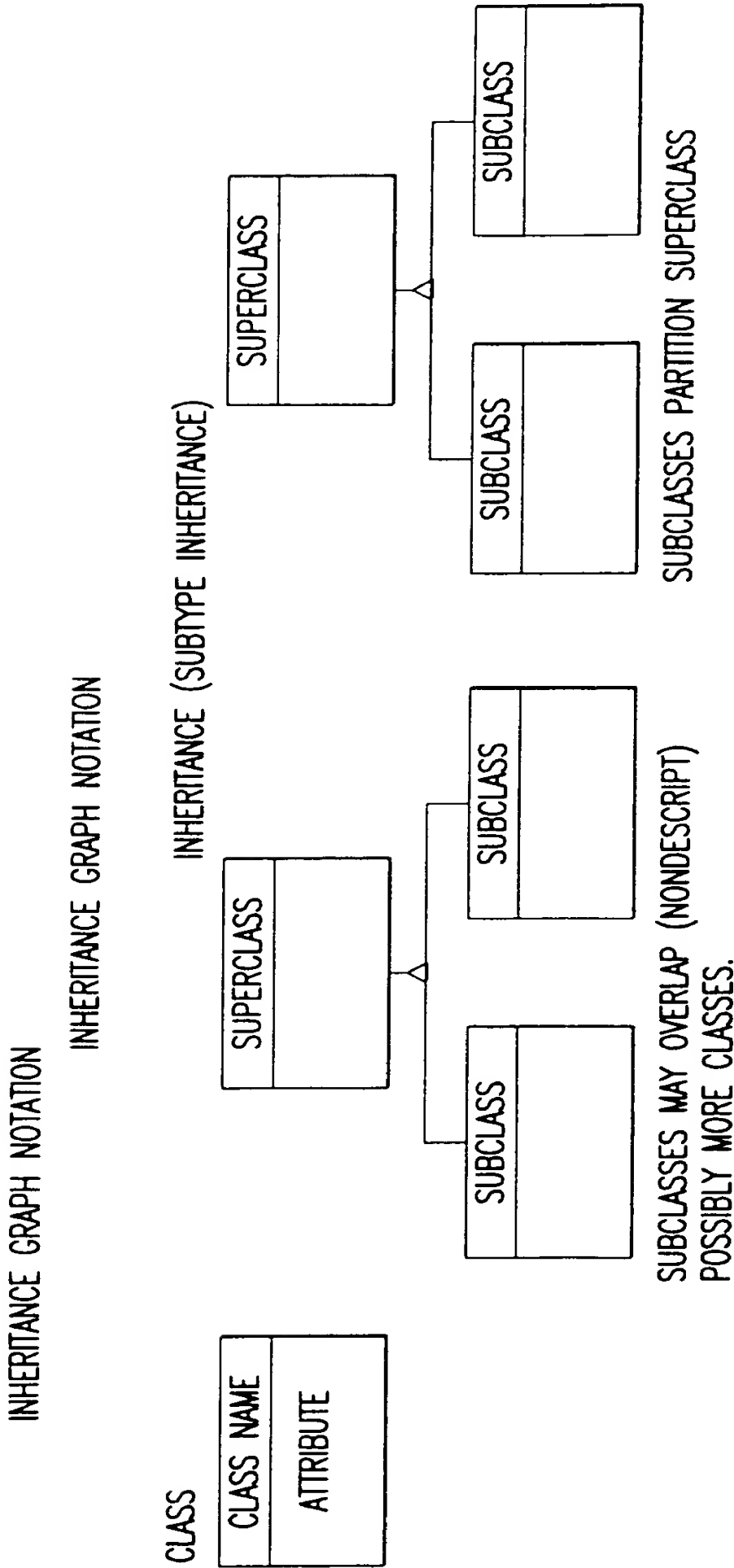


FIG.61

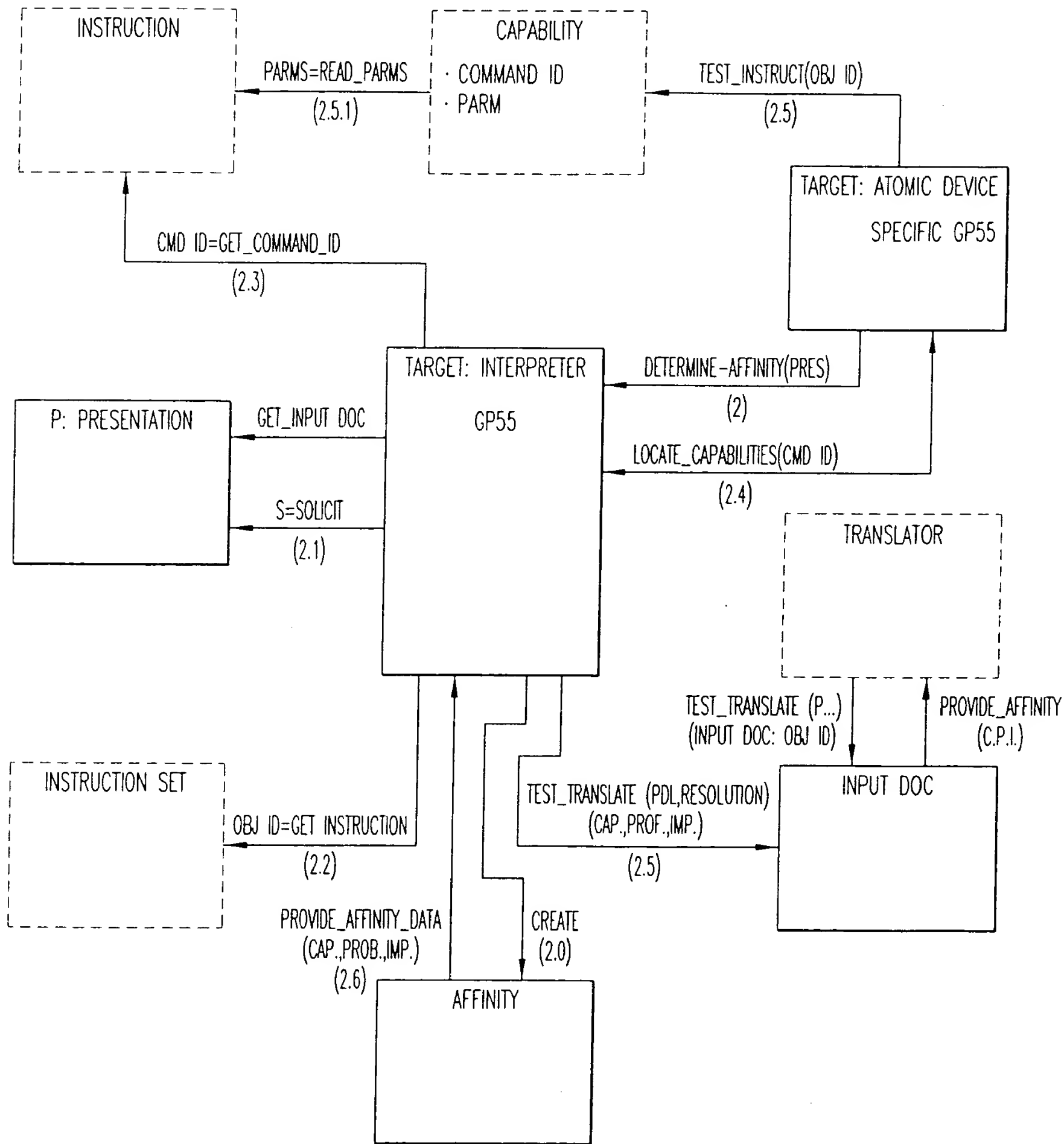


FIG.62

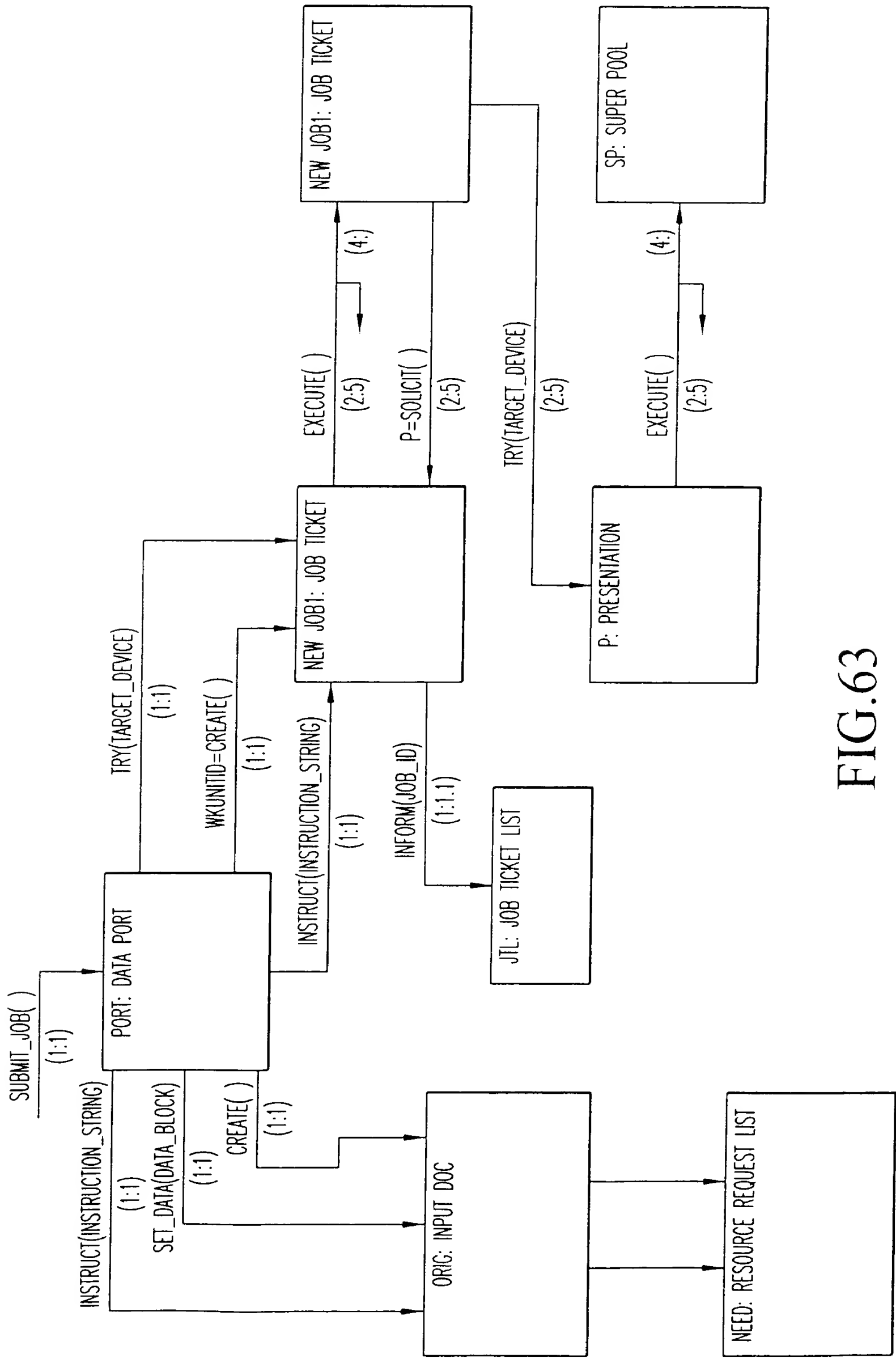


FIG.63

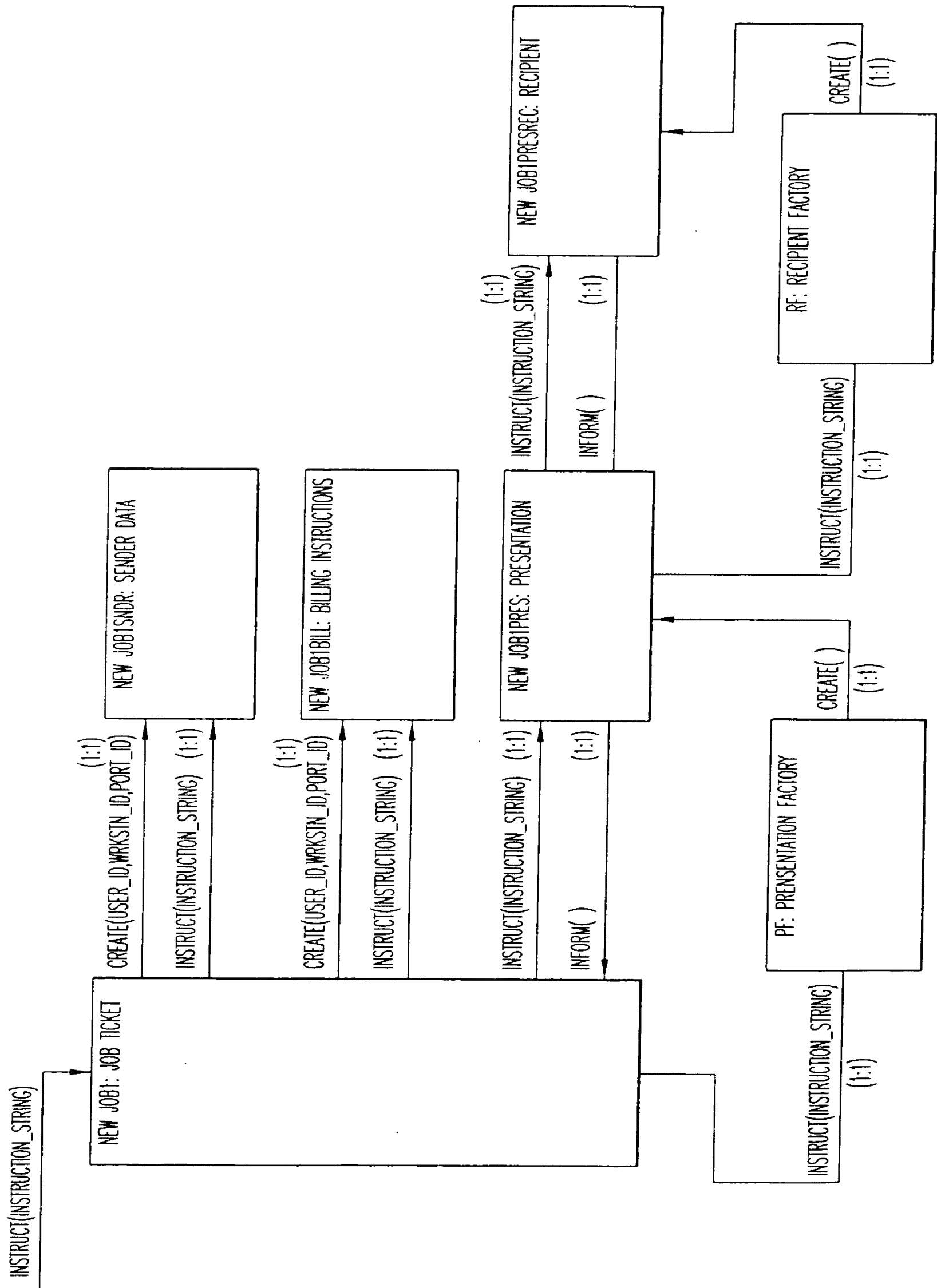


FIG.64

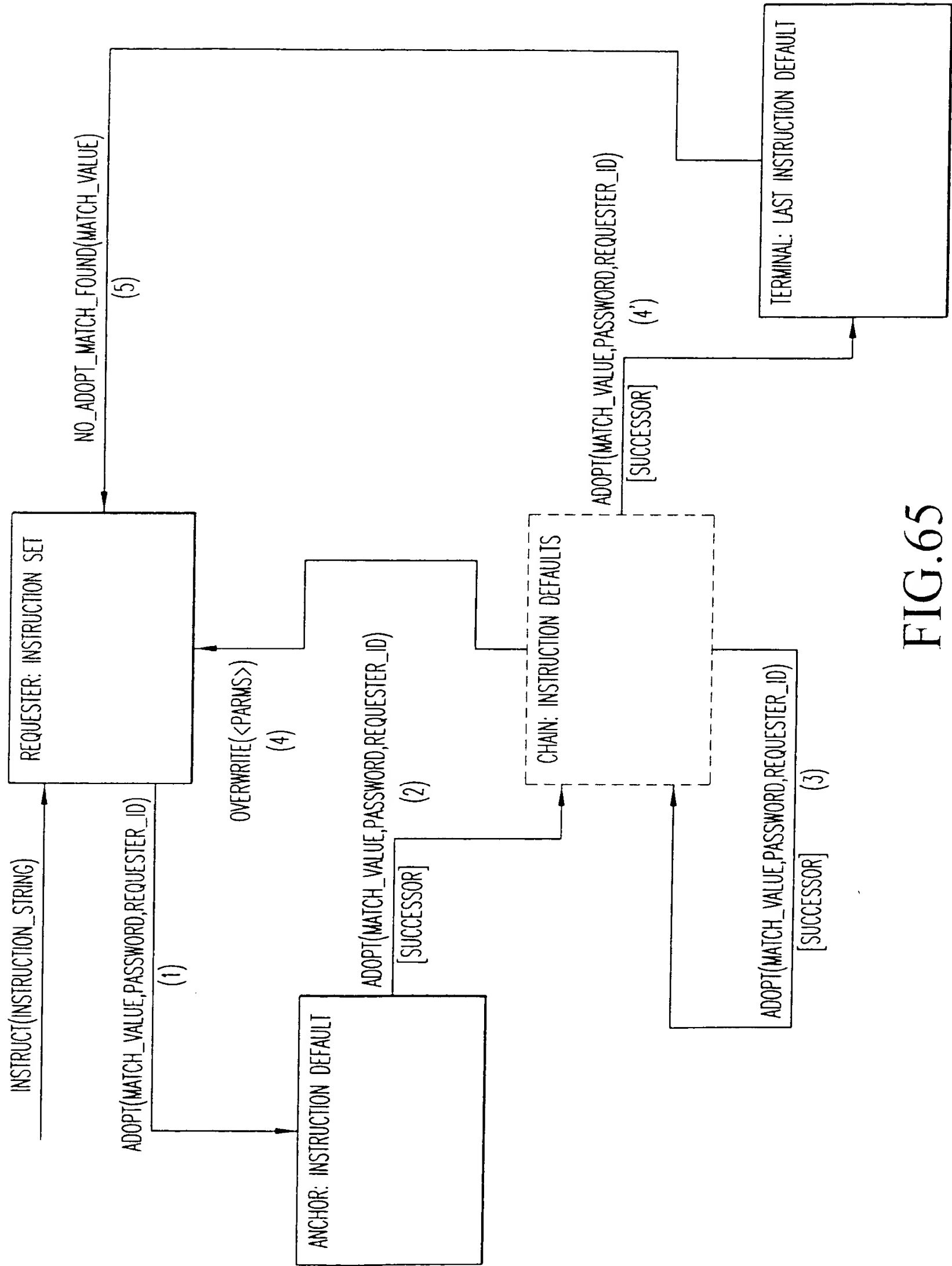


FIG.65

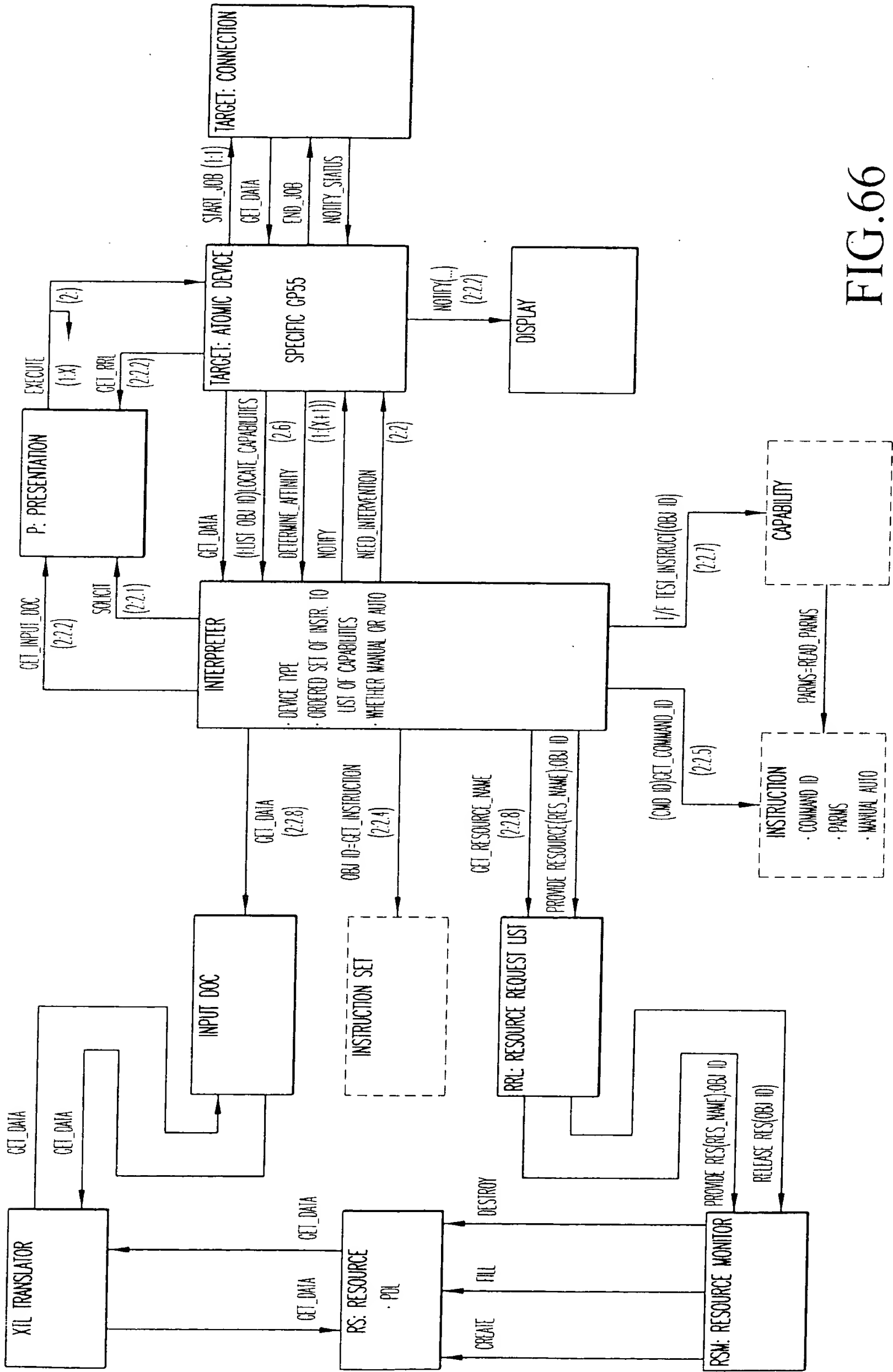


FIG.66

Enterprise

Recipients

Jeff Robinson

Add

Remove

Include Self

Edit

Duplicate

About you..

Presentations

Print Options

Fax Options

Archive Options

☐ Archive This Job

Special Instructions: ☐ Hold for Operator Review

Submit

Template: default.tpl

Load... Save As...

Cancel

Help

Enterprise

Copyright 1996-0998, Sper Products Corp. All rights reserved.

FIG.67

Recipient Information

Personal Company Shipping

☐ Print to this Recipient ☐ Fax to this Recipient

Name

Division

Mail Stop

E-Mail

Office Phone:

Fax Number

Cellular Number

Pager Number

OK Cancel Help

FIG.68

Recipient Information

Personal

Company

Shipping

Name

Address 1

Address 2

City

State / Province

Zip / Postal Code

Country

Phone

Fax Number

OK

Cancel

Help

FIG.69

Recipient Information

Personal Company Shipping

Carrier

- ☒ Pickup
- ☐ Federal Express
- ☐ UPS
- ☐ Mail
- ☐ DHL
- ☐ Airborne Express
- ☐

Priority

- ☒ Priority Overnight
- ☒ Standard Overnight
- ☒ 2nd Day
- ☒ Ground

OK Cancel Help

FIG.70

About You...

Personal

Company

Billing

Name

Division

Mail Stop

E-Mail

Office Phone:

Fax Number

Cellular Number

Pager Number

OK

Cancel

Help

FIG.71

About You...

Personal

Company

Billing

Name

Address 1

Address 2

City

State / Province

Zip / Postal Code

Country

Phone

Fax Number

OK

Cancel

Help

FIG.72

The image shows a software dialog box titled "About You...". It has three tabs: "Personal", "Company", and "Billing". The "Billing" tab is currently selected. Inside the dialog, there are several input fields and radio buttons. The "Credit Card" radio button is selected. Below it, there are fields for "Card Number", "Expiration Date", "Authorized Name", and "Maximum Amount to be Charged \$". There are also radio buttons for "Cash on Delivery" and "Point of Sale". At the bottom of the dialog are three buttons: "OK", "Cancel", and "Help".

About You...

Personal Company Billing

☒ Account #

☐ PO #

☒ Credit Card

Card Number

Expiration Date

Authorized Name

Maximum Amount to be Charged \$

☐ Cash on Delivery

☐ Point of Sale

OK Cancel Help

FIG.73

Print Options

Job Page Finishing Scheduling

Device Kodak IS70 Load

Sets Per Recipient 1

Collate None

Duplex None

Job Separator No

Form Name None

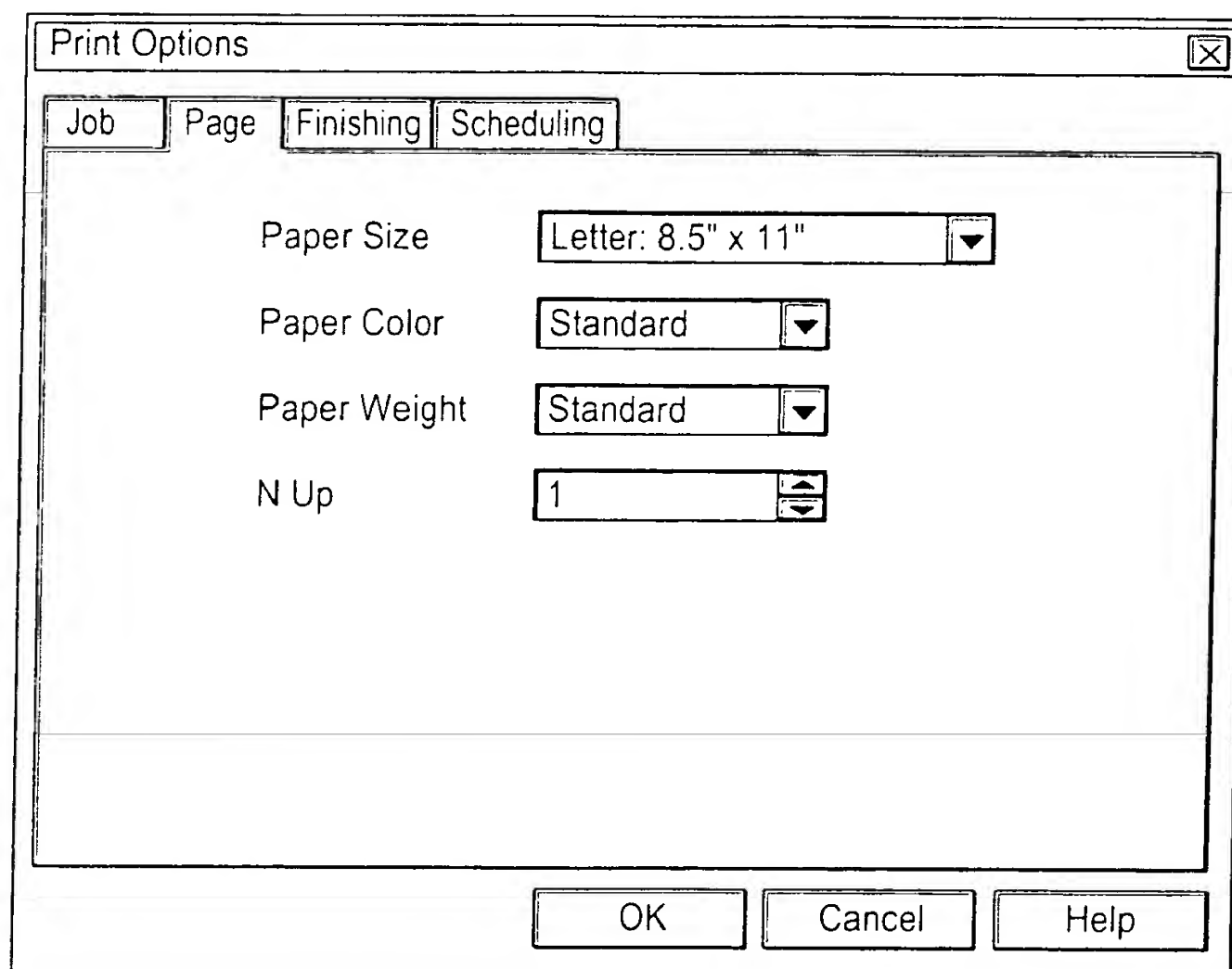
Output Type

☒ Black & White

☐ Color

OK Cancel Help

FIG.74



The image shows a 'Print Options' dialog box with a title bar containing a close button. Below the title bar are four tabs: 'Job', 'Page', 'Finishing', and 'Scheduling'. The 'Finishing' tab is currently selected. The main area of the dialog contains four settings, each with a label and a control element:

- Paper Size:** A dropdown menu showing 'Letter: 8.5" x 11"' with a downward arrow.
- Paper Color:** A dropdown menu showing 'Standard' with a downward arrow.
- Paper Weight:** A dropdown menu showing 'Standard' with a downward arrow.
- N Up:** A numeric input field showing '1' with up and down arrow buttons.

At the bottom of the dialog are three buttons: 'OK', 'Cancel', and 'Help'.

FIG.75

Print Options

Job

Page

Finishing

Scheduling

Staple

None

Stitching

None

Binding

None

Covers

None

Drilling

None

Folding

Z-Fold

Cutting

None

OK

Cancel

Help

FIG.76

Print Options

Job Page Finishing Scheduling

Send Time

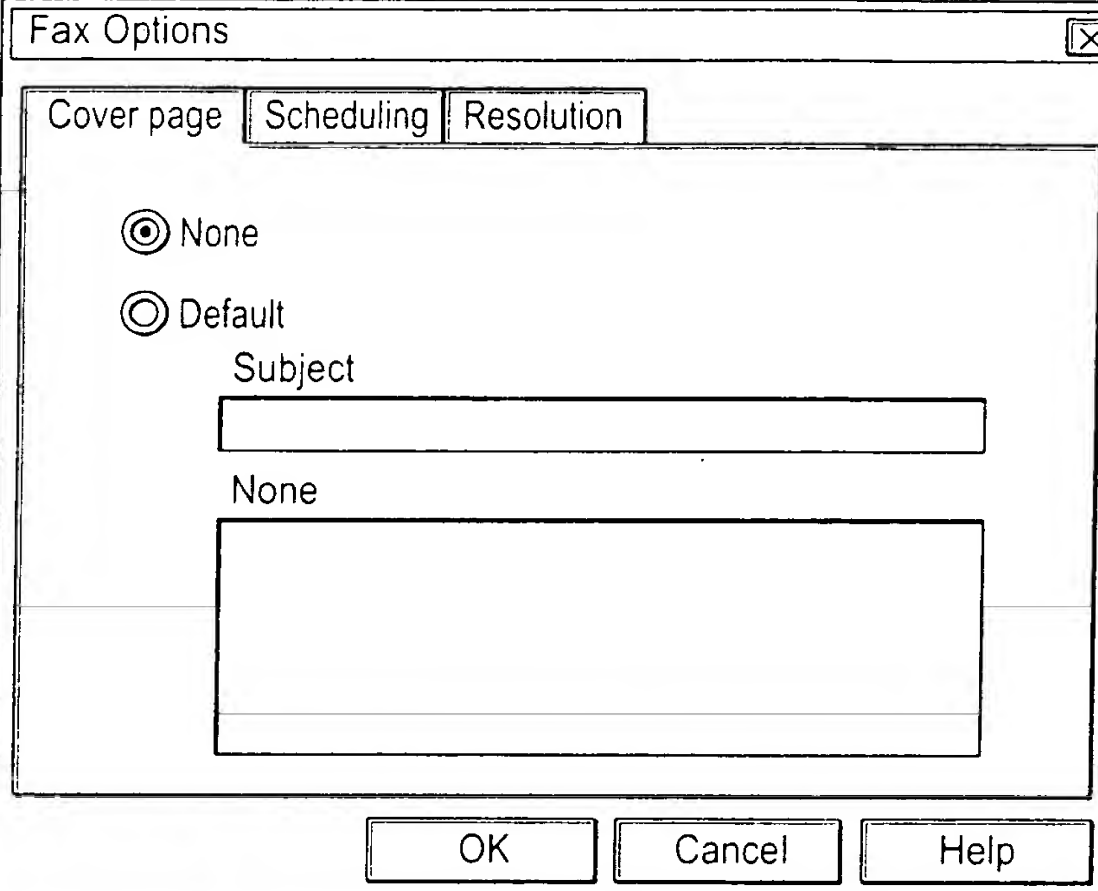
☒ Immediately
☐ Wait to Send

Date 7/14/98

Time 10:21 AM

OK Cancel Help

FIG.77



The image shows a 'Fax Options' dialog box with a standard Windows-style title bar and a close button. It contains three tabs: 'Cover page', 'Scheduling', and 'Resolution'. The 'Cover page' tab is active. Inside this tab, there are two radio buttons: 'None' (which is selected) and 'Default'. Below the 'Default' radio button is a text input field labeled 'Subject'. Below the 'Subject' field is another text input field labeled 'None'. At the bottom of the dialog box are three buttons: 'OK', 'Cancel', and 'Help'.

Fax Options

Cover page Scheduling Resolution

☒ None

☐ Default

Subject

None

OK Cancel Help

FIG.78

Fax Options

Cover page

Scheduling

Resolution

Send Time

☒ Immediately

☐ Wait to Send

Date

7/14/98

Time

10:21 AM

OK

Cancel

Help

FIG.79

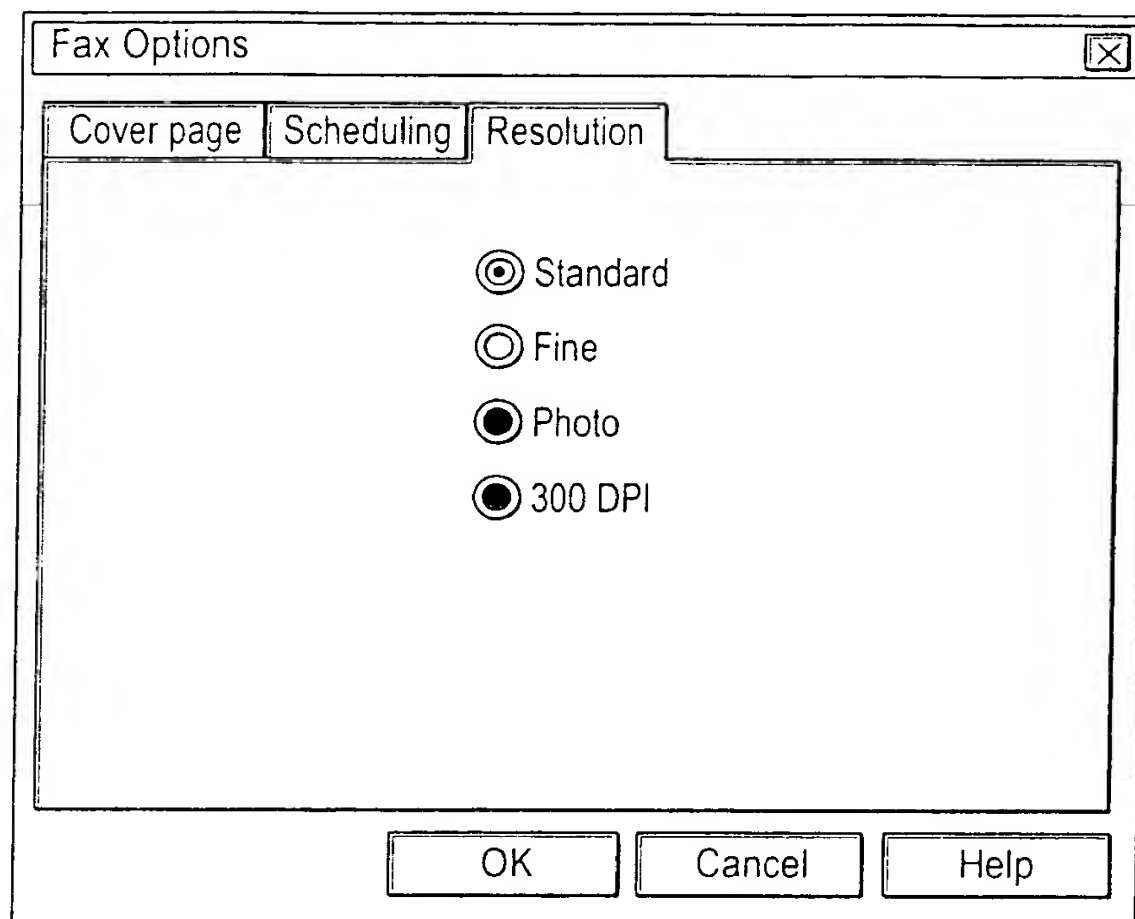


FIG.80

Archive Options

Format Adobe PDF

Archive Type

☒ Permanent Archive

☐ Removable Archive

Zip Disk

OK

Cancel

FIG.81